Notice of Allowability	Application No.	Applicant(s)	
	09/687,009	HARPER, JOHN	
	Examiner	Art Unit	
	Ian N. Moore	2661	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1. ☒ This communication is responsive to *9/15/2005*.

2. ☒ The allowed claim(s) is/are *1-12, 15-17,19-30,32-33, and 35-37 which have been renumbered as 1-32*.

3. ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
   a) ☐ All b) ☐ Some* c) ☐ None of the:
      1. ☐ Certified copies of the priority documents have been received.
      2. ☐ Certified copies of the priority documents have been received in Application No. _____ .
      3. ☐ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

   * Certified copies not received: _____.

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.
THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.

4. ☐ A SUBSTITUTE OATH OR DECLARATION must be submitted. Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL PATENT APPLICATION (PTO-152) which gives reason(s) why the oath or declaration is deficient.

5. ☒ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
   (a) ☐ including changes required by the Notice of Draftsperson's Patent Drawing Review (PTO-948) attached
      1) ☐ hereto or 2) ☐ to Paper No./Mail Date _____.
   (b) ☒ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date 1-10-06.

   Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).

6. ☐ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Attachment(s)

1. ☐ Notice of References Cited (PTO-892)
2. ☐ Notice of Draftperson's Patent Drawing Review (PTO-948)
3. ☐ Information Disclosure Statements (PTO-1449 or PTO/SB/08), Paper No./Mail Date _____
4. ☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material
5. ☐ Notice of Informal Patent Application (PTO-152)
6. ☒ Interview Summary (PTO-413), Paper No./Mail Date *1-10-06* .
7. ☒ Examiner's Amendment/Comment
8. ☐ Examiner's Statement of Reasons for Allowance
9. ☒ Other *See Continuation Sheet*.

Chau T. Nguyen

CHAU NGUYEN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2600

Continuation of Attachment(s) 9. Other: amended FIG. 1,9; & amended claims.

DETAILED ACTION

EXAMINER'S AMENDMENT

1. An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

Authorization for this examiner's amendment was given in a telephone interview with Cindy S. Kaplan on December 14, 2005.

The application has been amended as follows:

- In the **drawings, Figure 1,** label "**R**" has been replaced with "**Router**", label "S" has been replaced with "**Source**", and label "**D**" has been replaced with "**Destination**". (*See attached Figure*)
- In the **drawings, Figure 9,** label "**R**" has been replaced with "**Root Node**", a new label "**parent node**" has been added to label "**72**", a new label "**Subtree of a node**" has been added to label "**76**", and a legend "**—— Shortest Path**" has been added. (*See attached Figure*)
- **Claims 1,2,3, 7, 9, 15, 19,20, 22, and 29** have been amended (*See attach for the marked up claims*)

Allowable Subject Matter

2. Claims 1-12, 15-17,19-30,32-33, and 35-37 are allowed.

Conclusion

3. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Ian N. Moore whose telephone number is 571-272-3085. The examiner can normally be reached on 9:00 AM- 6:00 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Chau Nguyen can be reached on 571-272-3126. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

INM
1-10-06

Claim 1 (currently amended): A method for performing route calculations in a link state routing protocol at a node within a computer network, the method comprising:

receiving new route information at the node;

evaluating existing routes of the node before recalculating and modifying routes to determine if said new route information improves at least one of the existing routes or at least one of the existing routes is made worse or lost;

recalculating routes and modifying a routing table for the node only when said new route information improves at least one of the existing routes or at least one of the existing routes is made worse or lost, wherein recalculating routes comprises modifying information about links within the network without examining each of the nodes within the network; and

upon losing one of the existing routes:

initializing a best cost;

finding a neighbor node's link information by traversing a link state packet ~~an LSP~~;

calculating a neighbor cost of reaching another node via the neighbor node; and

setting the best cost to the neighbor cost if the neighbor cost is less than the best cost;

~~wherein recalculating routes comprises modifying information about links within the network without examining each of the nodes within the network~~

wherein only a parent node sending the new route information is used in recalculating routes if the new route information improves existing routes.

Claim 2 (currently amended): The method of claim 1 further comprising receiving a new link state packet with information about the node's path to a root node and wherein the node's route to the root node is improved and further comprising evaluating the node's neighbor nodes.

Claim 3 (currently amended): The method of claim 1 further comprising receiving a new link state packet with information about the node's path to a root node and wherein the node's route to the root node has worsened and further comprising evaluating the node's path to the root node.

Claim 4 (original): The method of claim 3 wherein nodes contained within a subtree containing the node are scrapped and the routes to all nodes in the subtree are re-evaluated.

Claim 5 (original): The method of claim 1 wherein recalculating existing routes comprises implementing equal-cost path splitting.

Claim 6 (original): The method of claim 5 wherein the new route information improves existing routes and only a parent node sending the new route information is used in recalculating routes.

Claim 7 (currently amended): A method for performing route calculations in a link state routing protocol at a node within a computer network, the method comprising:

receiving new route information at the node;

evaluating existing routes of the node before recalculating and modifying routes to determine if said new route information improves at least one of the existing routes or at least one of the existing routes is made worse or lost;

recalculating routes and modifying a routing table for the node only when said new route information improves at least one of the existing routes or at least one of the existing routes is made worse or lost, wherein recalculating routes comprises modifying information about links within the network without examining each of the nodes within the network; and

upon losing one of the existing routes:

initializing a best cost;

finding a neighbor node's link information by traversing a link state packet;

calculating a neighbor cost of reaching another node via the neighbor node; and

setting the best cost to the neighbor cost if the neighbor cost is less than the best cost;

wherein~~The method of claim 5 wherein the new route information worsens existing routes and~~ a parent node sending the information is no longer considered a parent node by said node if the new route information worsens existing routes.

Claim 8 (original): The method of claim 1 wherein the computer network comprises greater than one hundred nodes.

Claim 9 (currently amended): The method of claim 1 wherein said node has lost its path to ~~another~~ a different node within the computer network.

Claim 10 (original): The method of claim 9 further comprising reattaching the node at a location within a remaining portion of a spanning tree.

Claim 11 (original): The method of claim 11 further comprising recalculating routes to all other nodes in a subtree of which the node is a root node.

Claim 12 (original): The method of claim 1 further comprising performing an incremental route recalculation for all nodes within the network that have received new link state information.

Claims 13 -14 (canceled).

Claim 15 (currently amended): A computer program product for performing route calculations in a link state routing protocol at a node within a computer network, comprising:

code that evaluates existing routes of the node before recalculating and modifying routes ~~when new route information is received~~ to determine if said new route information improves at least one of the existing routes or at least one of the existing routes is made worse or lost;

code that recalculates routes and modifies a routing table for said node only when said new route information improves at least one of the existing routes or at least one of the existing routes is made worse or lost, wherein said code that recalculates routes comprises code that modifies information about links within the network without examining each of the nodes within the network;

code that, upon losing one of the existing routes:

initializes a best cost;

finds a neighbor node's link information by traversing a link state packet ~~an LSP~~;

calculates a neighbor cost of reaching another node via the neighbor node; and

sets the best cost to the neighbor cost if the neighbor cost is less than the best cost; and

a computer-readable storage medium for storing the codes;

~~wherein said code that recalculates routes comprises code that modifies information about links within the network without examining each of the nodes within the network~~wherein only a parent node sending the new route information is used in recalculating routes if the new route information improves existing routes.

Claim 16 (original): The computer program product of claim 15 wherein the computer-readable medium is selected from the group consisting of CD-ROM, floppy disk, flash memory, system memory, hard drive, and data signal embodied in a carrier wave.

Claim 17 (original): The computer program product of claim 15 further comprising code that performs equal-cost path splitting.

Claim 18 (canceled).

Claim 19 (currently amended): A system for performing route calculations in a link state routing protocol at a node within a computer network, the system comprising a processor operable to evaluate existing routes of the node ~~when new route information is received~~ before recalculating and modifying routes to determine if said new route information improves at least one of the existing routes or at least one of the existing routes is made worse or lost, recalculate routes and modify a routing table for said node only when said new route information improves existing routes or existing routes are made worse or lost, and upon losing one of the existing routes initializes a best cost, finds a neighbor node's link information by traversing ~~an LSP~~ a link state packet, calculates a neighbor cost of reaching another node via the neighbor node, and sets the best cost to the neighbor cost if the neighbor cost is less than the best cost; wherein recalculating routes comprises modifying information about links within the network without examining each of the nodes within the network and wherein only a parent node sending the new route information is used in recalculating routes if the new route information improves existing routes; and memory for storing route information.

Claim 20 (currently amended): A system for performing route calculations in a link state routing protocol at a node within a computer network, comprising:

means for evaluating existing routes of the node ~~when new route information is received~~ before recalculating and modifying routes to determine if said new route information improves at least one of the existing routes or at least one of the existing routes is made worse or lost;

means for recalculating routes and modifying a routing table for said node only when said new route information improves existing routes or existing routes are made worse or lost, wherein means for recalculating routes comprises modifying information about links within the network without examining each of the nodes within the network;

means for initializing a best cost when one of the existing routes is lost;

means for finding a neighbor node's link information by traversing ~~an LSP~~ a link state packet;

means for calculating a neighbor cost of reaching another node via the neighbor node;

means for setting the best cost to the neighbor cost if the neighbor cost is less than the best cost; and

memory for storing route information;

~~wherein recalculating routes comprises modifying information about links within the network without examining each of the nodes within the network~~ wherein only a parent node sending the new route information is used in recalculating routes if the new route information improves existing routes.

Claim 21 (original): The system of claim 20 further comprising means for performing equal-cost path splitting.

Claim 22 (currently amended): A method for performing route calculations in a link state routing protocol at a root node within a computer network, the method comprising:

receiving new route information at the root node;

sorting nodes with new route information into order of cost;

evaluating changes in state;

evaluating routes before reattaching routes, if existing routes are improved, lost, or made worse;

reattaching routes at lowest cost point in a spanning tree; and

re-evaluating routes from reattached nodes; and

upon losing one of the existing routes:

initializing a best cost;

finding a neighbor node's link information by traversing ~~an LSP~~ a link state packet;

calculating a neighbor cost of reaching a node via the neighbor node; and

setting the best cost to the neighbor cost if the neighbor cost is less than the best cost;

wherein only a parent node sending the new route information is used in evaluating the routes if the existing routes are improved.

Claim 23 (previously presented): The method of claim 22 further comprising splitting traffic across more than one path if total cost is the same for each of the paths.

Claim 24 (previously presented): The method of claim 23 wherein evaluating changes in state comprises performing incremental route recalculation.

Claim 25 (previously presented): The method of claim 22 wherein sorting nodes comprises sorting nodes into order of cost from the root node.

Claim 26 (previously presented): The method of claim 22 further comprising leaving routes unchanged if the new route information has no effect on existing routes.

Claim 27 (previously presented): The method of claim 22 wherein existing routes are lost or made worse and further comprising re-evaluating a subtree of the root node.

Claim 28 (previously presented): The method of claim 27 further comprising splitting paths among equal cost routes.

Claim 29 (previously presented): The method of claim 1 wherein each node within the computer network is represented by a data structure comprising information about links to other nodes and cumulative cost of all links traversed from a root to the node.

Claim 30 (previously presented): The method of claim 1 wherein recalculating routes from the node comprises applying an incremental Dijkstra's algorithm to the node.

Claim 31 (canceled).

Claim 32 (previously presented): The method of claim 1 wherein said at least one of the existing routes is made worse and further comprising recalculating routes to all nodes in a subtree of the node.

Claim 33 (previously presented): The method of claim 1 wherein recalculating routes comprises recalculating routes at all nodes which have received new link state information and processing said nodes in increasing order of distance from a root node.

Claim 34 (canceled).

Claim 35 (previously presented): The method of claim 1 further comprising applying an incremental Dijkstra's algorithm to the root node only if said new route information improves or worsens at least one of the existing routes or at least one of the existing routes is lost.

Claim 36 (previously presented): The method of claim 35 further comprising applying equal-cost path splitting.

Claim 37 (previously presented): The method of claim 35 wherein the number of nodes examined is proportional to the log of the number of nodes within the network.

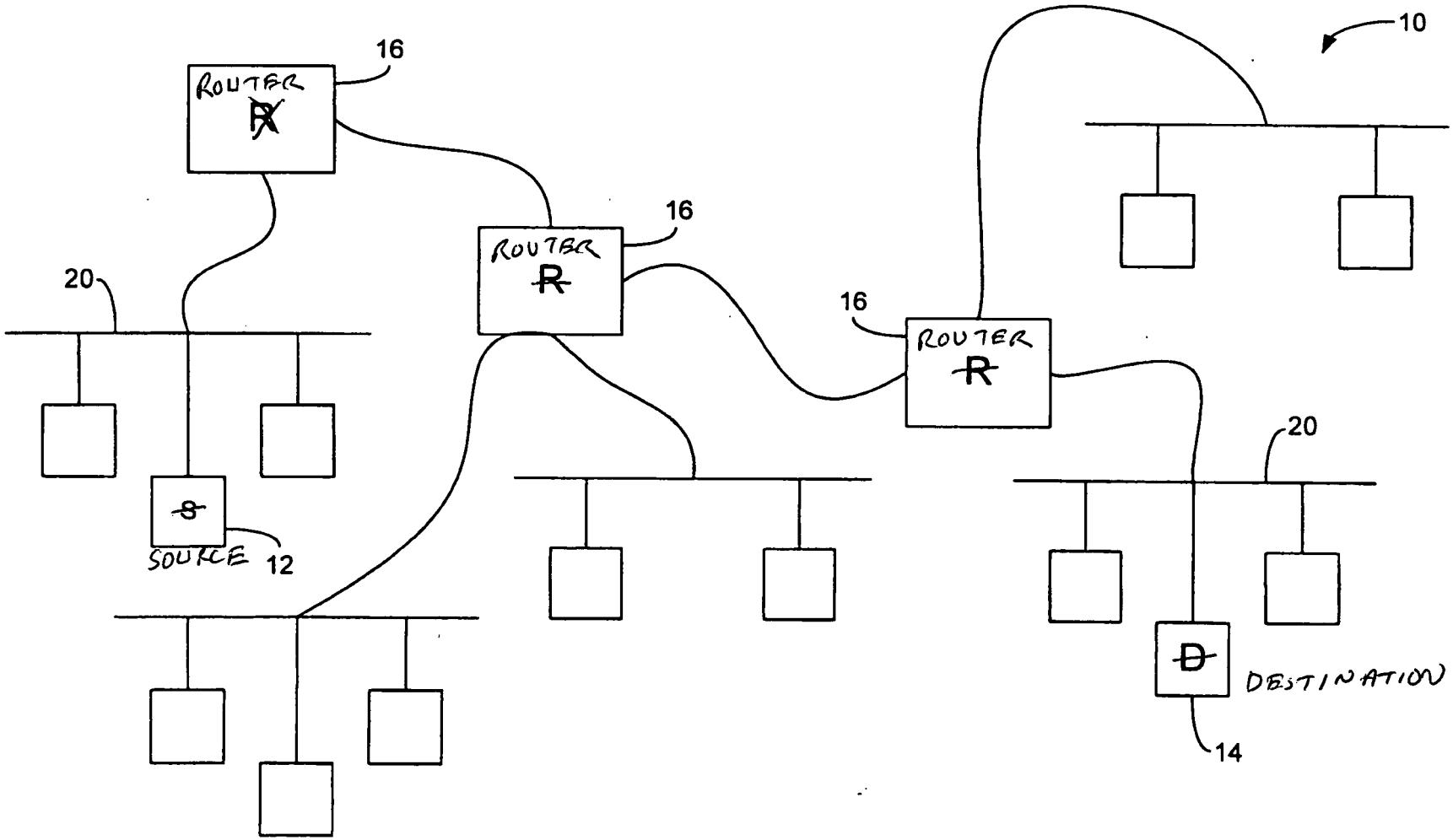

FIG. 1

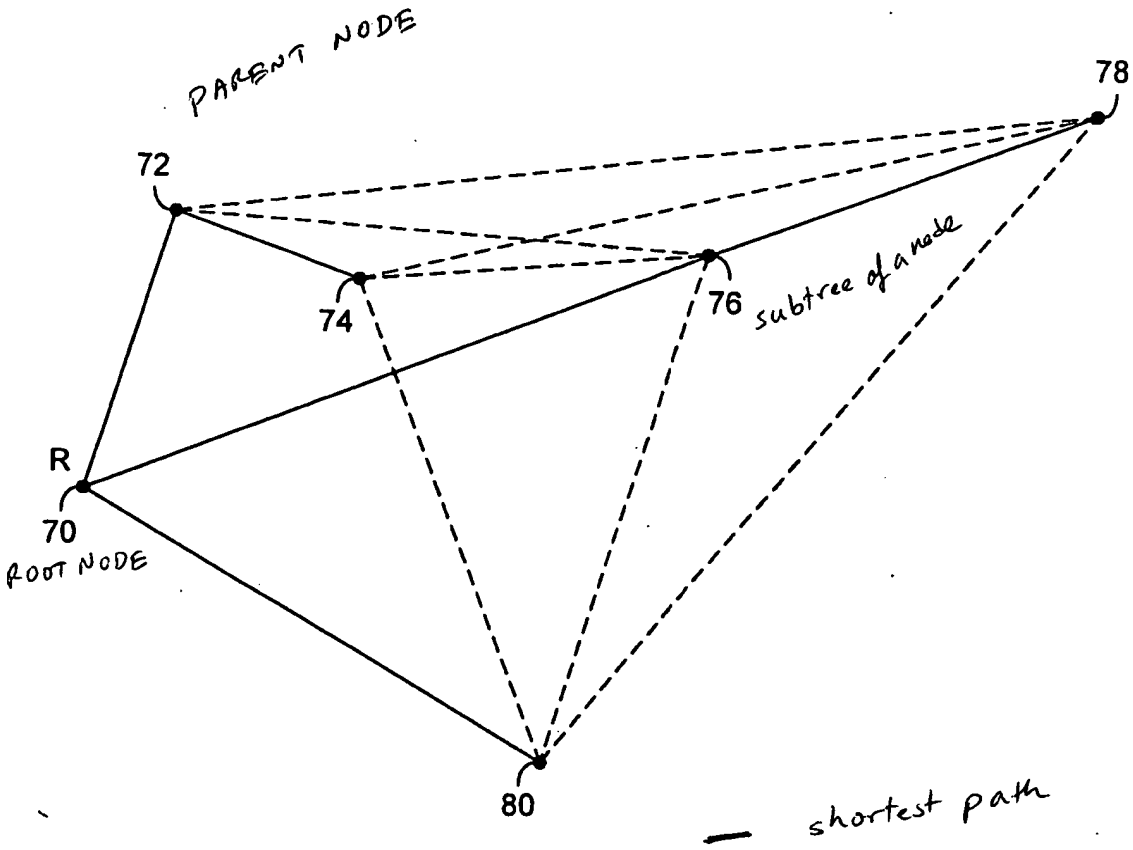
PARENT NODE
78
72
74
76
subtree of a node
R
70
ROOT NODE
80
shortest path

FIG. 9